

Remarks

Claims 16- 20 are pending. Claims 16- 20 are rejected. All rejections are respectfully traversed.

The invention is a method for distributed remote network monitoring (dRMON) in a LAN. For each of a plurality of ESs to be monitored, an associated dRMON agent in the form of executable code is deployed within the ES. The dRMON agents are configured to communicate with a dRMON proxy connected to the LAN, each dRMON agent implementing RMON functional groups but only capturing and analyzing packets transmitted and/or received by an associated ES. Periodically, the dRMON agents forward agent data including statistics and/or captured packets to the dRMON proxy. The forwarded agent data is combined at the dRMON proxy

Claim 16 is rejected under 35 U.S.C. 103(a) as being unpatentable over Raab, et al., (U.S. 6,047,321 – Raab”), in view of Desai, et al., (U.S. 5,781,703 – “Desai”), in further view of Engel, et al., (U.S. 6,115,393 – “Engel”). The rejection is respectfully traversed.

Raab describes probe-based remote monitoring system (RMON). The probe described by Raab is a “bump in the wire,” meaning a physical device configured to be inserted in communication lines between other devices to monitor communications between the other devices.

As such, the probe includes communications ports for coupling to the communication lines, and monitoring equipment, see, e.g., col. 4, lines 17-28, below:

20 probe 400. More specifically, a port (e.g., port 407) on the switch may be connected to a port (e.g., port 408) on the probe. Another port (409) on the probe, in turn, is coupled to a network device such as device 1. Probe 400 includes circuitry for repeating data packets between the switch and the network devices coupled to the probe.

25 The probe 400 utilizes internal bypass circuitry in promiscuously monitoring the communications medium coupling network devices 1 and 3 to the probe. The probe promiscuously monitors all traffic between the hub 200 and switch 220 destined for or received from either network device 1 or network device 3. All data traffic is captured and

There, Raab teaches coupling a physical probe between a switch and a device. In contrast, claimed is deploying, within each of a plurality of ESs to be monitored, executable code comprising an associated dRMON agent. While the Examiner should have understood that a dRMON *agent* is executable code, i.e. software, the Applicants have amended claim 16 herein for clarification. The Examiner, at paragraph 4 of the rejection, see below, relies on Raab to teach dRMON agents by improperly equates a physical RMON probe *device* with an *agent* :

4. ESs to be monitored, said dRMON agents implementing RMON functional groups but only capturing and analyzing packets that their native ES sends or receives, (e.g. col. 4, lines 5 - 57);

In fact, Raab teaches a single physical RMON probe implementing RMON functional groups, which can never make obvious distributed (dRMON) agents, deployed in the devices to be monitored, implementing RMON functional groups, as claimed.

Beyond the function of remote monitoring of networked devices, there are no similarities shared by the invention and Raab. The probes described by Raab can never be used to make obvious dRMON agents, as claimed.

The same is true for the Examiner's assertion at paragraph 5 of the rejection, below:

5. on a periodic basis having the dRMON agents forward statistics and/or captured packets to said dRMON proxy, existing somewhere on the LAN, (e.g. col. 4, lines 5 – 57); and

The Examiner's assertion that Raab teaches forwarding statistics or captured packets is contrary to what is taught by Raab at col. 4. in particular, the

Examiner is directed to col. 4, lines 25-32, below:

25 pling network devices 1 and 3 to the probe. The probe
promiscuously monitors all traffic between the hub 200 and
switch 220 destined for or received from either network
device 1 or network device 3. All data traffic is captured and
subsequently saved, e.g., for some form of analysis or
30 statistical compilation. The probe analyzes those packets
according to, for example, the remote monitoring standards
RMON I or II. These standards promulgate, for example,

There, Raab teaches that the probe device, which is coupled between the communicating devices captures, saves and analyses all data traffic to and from the network devices 1 and 3. That can never make obvious distributed agents deployed in ESs forwarding agent data including statistics and/or captured packets to a dRMON proxy, as claimed. The Examiner is requested to point out exactly which part of Raab teaches agents or proxies, as claimed. Raab is useless for making the invention obvious.

At paragraph 6, the Examiner contradicts his assertion in paragraphs 4 and 5 by admitting that Raab in fact fails to teach dRMON agents deployed within ESs, and that communicate with a dRMON proxy. The Applicants agree with the Examiner's admission of the shortcomings of Raab and wonder

why Raab is referenced at all. The Examiner then points to Desai. However, Desai fails to cure the defects of Raab.

Desai describes intelligent agents in devices to be monitored. The intelligent agents forward monitoring data to a proxy controller in a server. However, the data forwarding by the intelligent agents is only in response to requests from the proxy controller, see, e.g., col. 5, line 65 – col. 6, line 9, below:

PERFORMANCE MONITORING FUNCTIONS

65

In the present invention, there are two ways to collect performance data from Intelligent Remote Agents 18: (1)

6

transmitting commands for a situation monitoring request to a particular Intelligent Remote Agent 18 on a particular computer system 12; or (2) transmitting commands for a report request to a particular Intelligent Remote Agent 18 on a particular computer system 12. In both cases, the commands are transmitted from the Proxy Controller 16 to the Intelligent Remote Agent 18 and the response is returned by the Intelligent Remote Agent 18 to the Proxy Controller 16 and Data Server 14.

In Desai, intelligent agents forward data only in response to transmitted commands. In contrast, claimed is forwarding, periodically by the dRMON agents, agent data including statistics and/or captured packets to said dRMON proxy. Desai fails to cure the defects of Raab.

Engel is similar to Raab in that it describes a physical device configured to be inserted in communication lines between other devices to monitor communications between the other devices, see Engel, col. 6, lines 52-65, below:

Network Monitor 10 (referred to hereinafter simply as Monitor 10) is the data collection module which is attached to the LAN. It is a high performance real time front end processor which collects packets on the network and performs some degree of analysis to search for actual or potential problems and to maintain statistical information for use in later analysis. In general, it performs the following

A person of ordinary skill in the art would never confuse a physical processor inserted in communications lines between networked devices as in Engel and Raab, with dRMON agents deployed in devices to be monitored, as claimed. Further, the Examiner's assertion that Engel teaches dRMON agents communicating with a dRMON proxy is pure conjecture, because the Examiner's reference to col. 27 of Engel describes a monitor registering with a management workstation, which is the operator interface for the monitor, see col. 6, line 66 – col. 7, line 9. There is an agent resident in the monitor to communicate with the management workstation, but it should be understood that the monitor agent can never be a dRMON agent deployed in devices to be monitored. The Engel agent is deployed in a device doing the monitoring. Therefore, the rejection of claim 16 based on the combination of Raab, Desai and Engel should be reconsidered and withdrawn.

Claims 17 and 18 are rejected under 35 U.S.C. 103(a) as being unpatentable over Raab, Desai and Engel, and in further view of Dobbins, et al. (U.S. 5,790,546 – “Dobbins”).

Dobbins describes a method for secure fast packet switching in which MAC ID of devices sending or receiving packets across the network are recorded by an agents residing in switches in the network and associated with ports on the switches. Dobbins uses the information recorded by the agents to set up virtual connections or virtual LANS in the network in order to ensure varying QoS levels for different sets of devices on the network. As shown in Figures 7A and 7B, the agent resides in a switch, which would never be confused with an ES by a person of ordinary skill in the art. The agents described by Dobbins cannot perform their function if they resided on an ES

instead of a switch. The Examiner's assertion that Dobbins has anything to do with dRMON proxies or dRMON is pure conjecture. There is nothing in Dobbins that has anything to do with RMON as in Raab and Engel, or the command driven dRMON as in Desai. The same is true for claim 18.

Claim 19 is rejected under 35 U.S.C. 103(a) as being unpatentable over Raab, Desai and Engel, and in further view of Umetsu (U.S. 5,751,963).

Umetsu describes RMON, a network management system that includes a proxy agent node communicating with a plurality of network management nodes. This is a probe-based RMON system that teaches nothing about distributed RMON. In claim 19, the dRMON agents perform continual response time monitoring and forward monitoring results to the dRMON Proxy. Therefore, Umetsu can never describe dRMON agents performing continual response time monitoring or a dRMON proxy, as claimed.

Claim 20 is rejected under 35 U.S.C. 103(a) as being unpatentable over Raab, Desai and Engel, and in further view of Nugent (U.S. 6,076,131).

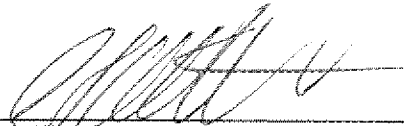
In claim 20, the executable code utilizes native OS APIs to gather information about the ES that could not be gathered via packet capture and analysis, said information being selected from the group consisting of: (1) Network protocol stack configurations and NIC configurations including problematic situations; (2) Application information including what protocols an application is bound to, to its manufacturer, version, file date and time, DLLs used and their versions; (3) System information such as memory, CPU, disk space, current resource utilizations; and (4) System performance

metrics. As to (1), the Examiner references Nugent at col. 9, lines 30-61. There, Nugent describes the protocol used to achieve routing resource reserve/release. There is absolutely no description of an agent utilizing native OS APIs to gather information about Network protocol stack configurations and NIC configurations including problematic situations, as claimed. At col. 9, lines 30-60, Nugent just describes the protocol source code. The Examiner is requested to specifically point out where Nugent describes an agent gathering information about the protocol. Further, as stated above with respect to claim 16, Engel never describes a dRMON agent, as claimed, and therefore is useless for making the invention obvious. The rejection should be reconsidered and withdrawn.

It is believed that this application is now in condition for allowance. A notice to this effect is respectfully requested. Should further questions arise concerning this application, the Examiner is invited to call Applicant's attorney at the number listed below. Please charge any shortage in fees due in connection with the filing of this paper to Deposit Account 50-6350.

Respectfully submitted,
3Com Corporation,

By



Andrew J. Curtin
Attorney for the Assignee
Reg. No. 48,485

350 Campus Drive
Marlborough, MA 01752
Telephone: (508) 323-1330
Customer No. 56436